Expert Systems With Applications 277 (2025) 127276

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Broad information diffusion modelling for sharing link click prediction using knowledge graphs

Xiangjie Kong [a,b], Can Shu [a], Lingyun Wang [a], Hanlin Zhou [a], Linan Zhu [a,*], Jianxin Li [c]

[a] College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou, 310023, China
[b] Zhejiang Key Laboratory of Visual Information Intelligent Processing, Zhejiang University of Technology, Hangzhou, 310023, China
[c] School of Business and Law, Edith Cowan University, Perth, 6009, Australia

ARTICLE INFO

Dataset link: https://github.com/CanShu6/K-BID

Keywords:
Knowledge graph
Information diffusion
Recommendation
Graph neural network

ABSTRACT

In the new media era, users actively share and diffuse information across social networks, creating complex patterns of broad information diffusion (BID) that differ significantly from traditional recommendation scenarios. Existing models are primarily designed for deep information diffusion (DID) with sequential cascades and struggle to address BID challenges, including the sparse graph structure, weak temporal correlation, and ambiguity in user preferences. To bridge this gap, we propose K-BID, a knowledge-driven framework tailored for BID scenarios. K-BID integrates semantic and social graph information through a two-phase 'Match & Rank' approach. The matching phase retrieves candidate voters using social relationships and personalized preferences, whereas the ranking phase refines predictions by modelling temporal dynamics. Experiments on real-world datasets demonstrate the superiority of K-BID over state-of-the-art methods, achieving significant improvements of 14.02%, 16.80%, and 16.99% in Precision, MRR, and AUC respectively, for the 'Soc.' objective with $K = 5$. Our work advances the understanding of BID scenarios and offers a practical solution for optimizing information dissemination in social platforms.

1. Introduction

Driven by similar interests or social relationships, users in online social platforms actively engage in sharing and diffusing diverse Internet information (Wu, Zhong, & Ye, 2023a; Zhai, Liu, Yang, & Xiao, 2023). This covers a wide range of items, such as short videos and news sourced from E-commerce, TikTok, Reddit and Weibo (Khaledian, Nazari, Khamforoosh, Abualigah, & Javaheri, 2023; Wu, Sun, & Zhang, 2023b). Considering the context of information diffusion, recommendation models should capture the evolving preferences of users and facilitate the dissemination and adoption of shared items within communities (Ma et al., 2024; Zhang, Kong, Shen, Wu and Qu, 2024).

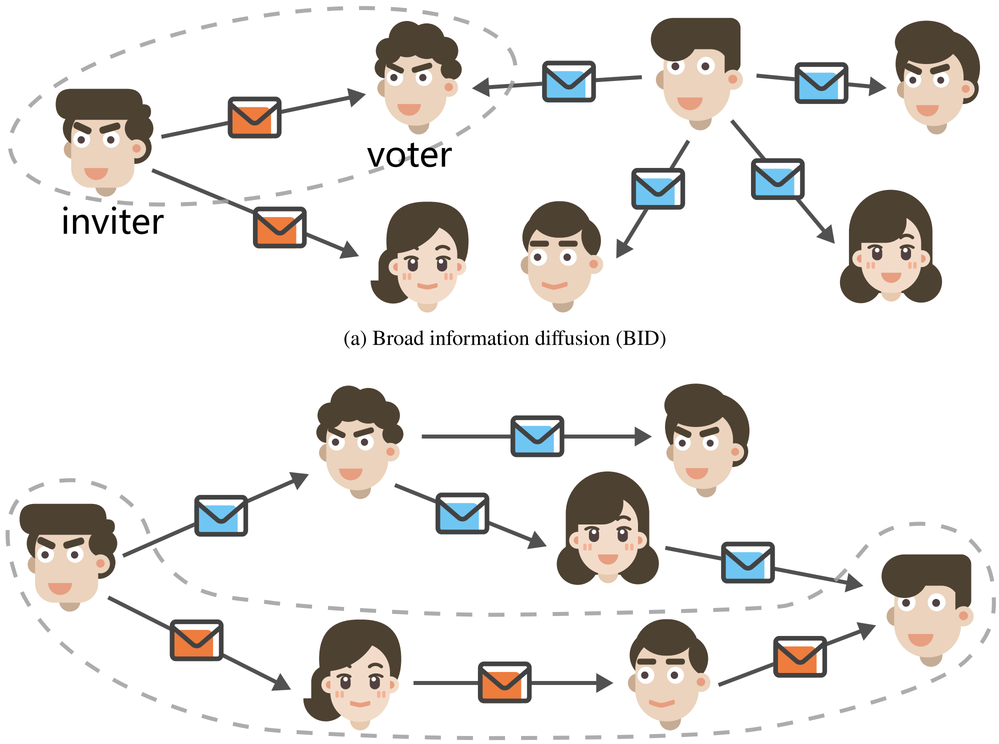
For a specific instance of item sharing, the user acting as an **'inviter'** posts or forwards the item information, whereas the user acting as a **'voter'** clicks on the item link shared by the inviter, thus constituting a single-step information diffusion. As shown in Fig. 1, we categorize the concept of information diffusion into broad information diffusion (BID) and deep information diffusion (DID). The fundamental distinction lies in the depth of the information cascade. Compared with DID, BID tends to establish ego-groups that are centred on inviters, failing to construct temporal sequence data with the first inviter as a starting point. In contrast to classical personalized recommendation tasks, we innovatively study the voter-oriented prediction task in BID, enabling item links shared by inviters to reach appropriate user groups.

Accordingly, there are challenges result from by the particularity of BID, in comparison with DID and classical recommendation:

- **The temporal correlation of BID is weak.** For the information diffusion chain of DID shown in Fig. 1(b), the current user's forwarding behaviour is instigated by the preceding user, maintaining precise chronological order across multiple diffusion steps. In contrast, it is difficult for BID to generate continuous time series, and each diffusion step tends to be independent.
- **The graph structure of BID is sparser than DID.** Previous information diffusion-based models (Li et al., 2019; Wu, Li, Sun, & Wang, 2020; Wu et al., 2019) used to transform diffusion chains into multi-hop paths, thereby strengthening the graph-signal propagation between user nodes. Nevertheless, the shorter diffusion chains of BID cannot establish the high-order connectivity from one to multi-hop neighbours.

* Corresponding author.
E-mail addresses: xjkong@ieee.org (X. Kong), cshu@zjut.edu.cn (C. Shu), lywang@zjut.edu.cn (L. Wang), hanlinzhou@zjut.edu.cn (H. Zhou), zln@zjut.edu.cn (L. Zhu), jianxin.li@ecu.edu.au (J. Li).

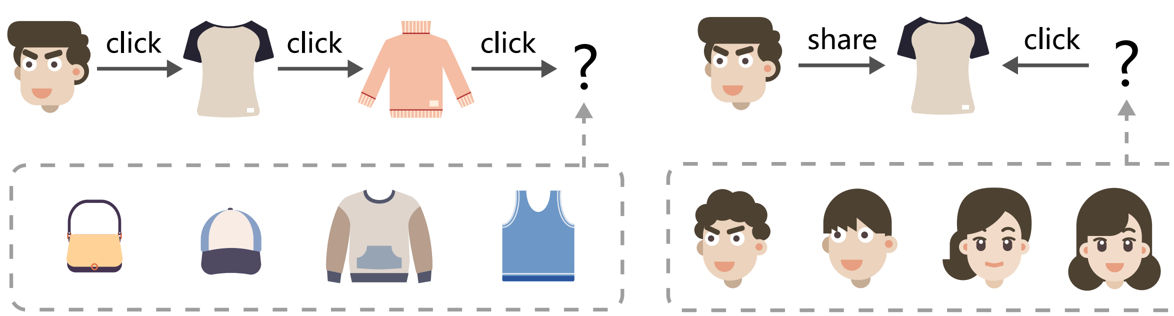
https://doi.org/10.1016/j.eswa.2025.127276
Received 28 November 2024; Received in revised form 13 March 2025; Accepted 13 March 2025
Available online 22 March 2025



(a) Broad information diffusion (BID)

(b) Deep information diffusion (DID)

Fig. 1. Different information diffusion. (a) BID means voters rarely forward the same information repeatedly, and the grey dashed circle represents one diffusion step. (b) DID means voters often forward the same information several times, and the grey dashed circle represents a diffusion chain.



(a) Classical recommendation

(b) Information diffusion

Fig. 2. Comparison of classical recommendation and information diffusion. (a) Given history user-item interactions, a classical recommendation model infers the next item most likely to be clicked. (b) Given an inviter and a shared item, our model infers which user will click the sharing link.

- **Preference ambiguity exists in information diffusion scenario.** Classical recommendation models (Lyu et al., 2022; Zhao et al., 2022) rely on personalized preferences to determine item presentation. However, discerning whether a voter clicks on a sharing link due to social contact or personal interests is imperative in BID. Fig. 2 illustrates their differences in prediction objectives.

As mentioned previously, the short diffusion paths in BID are sensitive to the issue of graph sparsity. Explicitly modelling node roles would significantly increase the complexity of model (Tong, Yuan, Jalili, Dong, & Sun, 2024; Xu & Dong, 2024). This could lead to overfitting, especially in areas of the network where data are sparse. We aim to maintain a balance between model complexity and generalizability to ensure that our model performs well. The focus on node influence and criticality can be more practical than explicitly modelling roles (Li, Fu, Yan, Zhao and Zeng, 2024; Zhao et al., 2024), especially in dynamic and evolving networks. TPC (Zhang, Wang, Liu and Wang, 2024) investigated the importance of node roles in different networks using various centrality measures, where each centrality parameter plays a role in social networks. However, the effectiveness of these measures varies depending on the characteristics of the network. Accurately determining node roles requires comprehensive and detailed data, including user behaviour and network structure. Given the data limitations and the complexity of explicitly modelling trust relationships, we opted to implicitly model roles through the available data. To address this, we construct knowledge graphs incorporating additional user and item attributes to enrich topological connections. Past researches focus on the deep diffusion scenario (de Souza & Durao, 2024; Kong, Mao, Wang, Liu, & Xu, 2021). However, thoroughly tracing the propagation of item information is not always feasible in the real world. Recent studies have identified several limitations and challenges in the current state of information diffusion and recommendation systems. For users with different levels of interaction, the added noise is sensitive to parameter changes, which will lead to the loss of user personalized information.

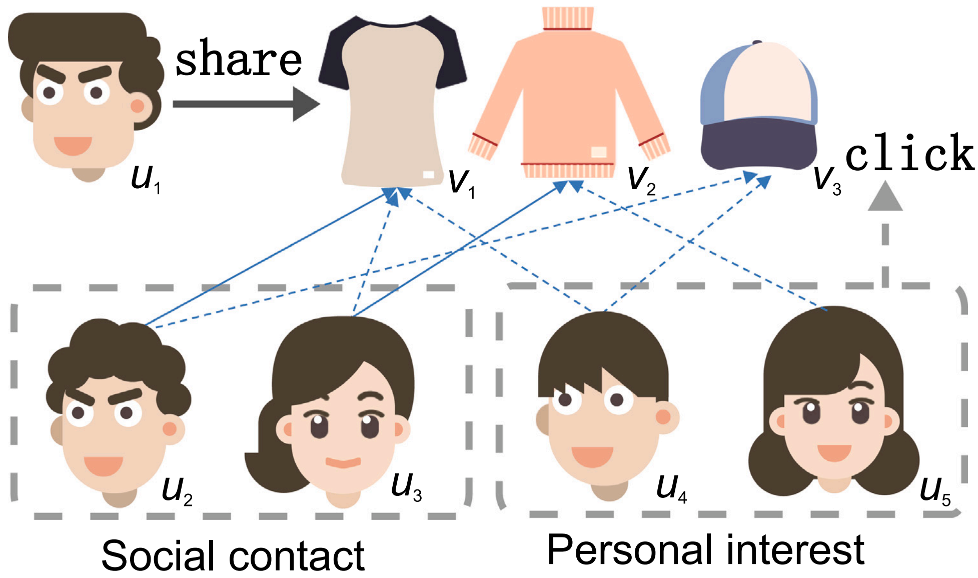


Fig. 3. A toy example illustrates the voter click process.

This may obscure the preferences captured by the model and affect the effectiveness of the recommendation (Jiangzhou, Songli, Jianmei, Lianghao, & Yong, 2024; Li, Guo, Liu and Yu, 2024). User interests can change over time, and a static model may not be able to adapt to these changes effectively (Zhou, Haq, Qiu, & Akbar, 2024). The lack of interpretability can make it difficult to explain the recommendations to users, which is important for user trust and acceptance of the recommendation system. Consequently, we focus on the broad diffusion scenario, drawing on algorithmic ideas from deep information diffusion to inform our approach.

Moreover, when complementing a quad (*inviter, shared item, shared time, ?*), it is essential to consider both the social relationships of inviter-voter pairs (**Factor 1**) and the matching preferences of item-voter pairs (**Factor 2**). By considering these factors, we can effectively learn user interests and higher-order collaborative semantics in the social domain.

Fig. 3 presents an example of a voter clicking on items, where the solid and dashed lines respectively represent the voter clicking on items due to the dominant influence of Factor 1 and Factor 2, respectively. Suppose that inviter u_1 and voters u_2 and u_3 have social connections, so that their clicking processes are closely related and influenced by the dominant Factor 1. When inviter u_1 shares items v_1, v_2, and v_3, voter u_2 clicks on item v_1 owing to social relationships Factor 1, even though they might not have been originally interested in it, and clicks on item v_3 owing to preference Factor 2. Similarly, voter u_3 clicks on item v_2 owing to social relationships Factor 1 and clicks on item v_1 owing to preference Factor 2. Meanwhile, voters v_3 and voter v_3 have no social relationships with inviter u_1, and their clicking processes are entirely influenced by preference Factor 2. This enables us to discover the inherent implicit item relationships in the actual interaction behaviours of users to enhance recommendation performance.

Motivated by the workflow of modern recommender systems (Zhu et al., 2022), we propose a knowledge-driven broad information diffusion model (K-BID). In production-level recommender systems, an item corpus is subjected to multiple rounds of matching and ranking processes, such as retrieval, rough sorting and fine sorting. In the matching phase, the model aims to retrieve potential voters by constructing social and collaborative knowledge graphs. By embedding these graphs using a semantic cross unit (SCU) and a graph attention network (GAT), the model can effectively capture the semantic interactions and the importance of different neighbours. The SCU crosses the embeddings of head entities and relations to capture their interactions, and the GAT aggregates neighbour information with attention scores to emphasize relevant neighbours. This dual approach allows the model to retrieve voters based on both social relationships and item preference, addressing the challenges of graph sparsity and preference ambiguity. In the ranking phase, the model refines the retrieved voters by incorporating a temporal prediction model. The knowledge retrieved in the matching phase is transferred to the ranking phase, where the model initializes the embeddings with pretrained weights from the knowledge graph. The model then captures the short-term preferences of voters using a Gated Recurrent Unit (GRU) and an attention mechanism, which aids in modelling the dynamic changes in user interests. Finally, the model combines the inviter embedding, shared item embedding, voter embedding, and short-term preference embedding to predict the likelihood of a voter clicking on the shared link. This comprehensive approach allows the model to make more accurate predictions by considering both social and preference factors in a unified manner.

The contributions of this paper are summarized as follows:

- We provide a formalized definition of the recommendation problem in broad information diffusion and identify the unique challenges that remain to be addressed.
- We propose a knowledge-driven broad information diffusion model (K-BID) composed of matching and ranking phases. This model integrates both social relationships and personalized preferences to enhance recommendation accuracy.
- We devise a semantic cross unit and a graph attention network to embed knowledge during the matching phase. In the ranking phase, we utilize a GRU to model short-term preferences and generate the final predictions.

The remainder of this paper is organized as follows: Section 2 covers related work, and Section 3 presents relative definitions and task formulations. Section 4 presents the matching phase in detail. Section 5 elaborates the K-BID of the framework and ranking phase. Section 6 describes the experimental setup and results analysis. Section 7 concludes this work and provides future directions.

2. Related work

2.1. Information diffusion in social networks

The process of information diffusion involves a group of users and propagated items. Thus, information diffusion chains can be appended to existing social relations within communities. Based on this extended social network, DiffNet++ (Wu et al., 2020) combines user-user relationships and user-item interactions to emphasize both social influence and personalized interests. In contrast to the aforementioned objectives, NFM (He & Chua, 2017) integrates neural networks to handle nonlinear relationships and enable higher-order feature interactions for refined predictions. Similarly, NDM (Yang et al., 2021) employs a multi-head attention mechanism to embed inviters and voters into separate feature spaces, quantifying the influence of previous voters on the next prospective voter. CKE (Zhang, Yuan, Lian, Xie, & Ma, 2016) integrates knowledge graphs to enhance recommendation systems by blending entities and relationships for collaborative filtering. MGDCF (Hu, Hooi, Qian, Fang, & Xu, 2024) uses Markov diffusion to learn the distance between the user and the item and predict the user's preference for the item. GDSREC (Chen, Xin, Liang, He, & Liu, 2023) models the social relationships among users by vectorizing the score offset and integrating it into the user and object representation learning. While effective for personalized recommendations, these models ignore the social context and external knowledge, leading to limitations in capturing user interests. In addition, these methods cannot distinguish the difference in auxiliary information for new users without any interaction, resulting in redundant and noisy information. We combine social relationships and semantic knowledge to enhance the recommendation accuracy and handle data sparsity.

2.2. Knowledge-driven recommendation

Knowledge graph represents and connects real-world concepts and entities in the form of (*head*, *relation*, *tail*). In the recommendation scenario, a vast number of users and items can be systematically organized into a heterogeneous directed graph (Guo et al., 2022; Kong, Chen, Li, Bi, & Shen, 2024a; Kong et al., 2022). In addition, node attributes and semantic information are also integrated into GNN for better performance, for example, ASNE (Liao, He, Zhang, & Chua, 2018), and STGNN (Kong, Shen, Wang, Shen, & Fu, 2024b). The vanilla message passing process only aggregates information from first-order neighbour nodes and is unable to leverage semantic correlations of non-first-order contexts. CGAT (Liu et al., 2023) addresses this issue by employing random walk to sample multi-hop neighbours and capturing their semantic correlations by GRU. KGAT (Wang, He, Cao, Liu and Chua, 2019) further improves graph-based recommendation by distinguishing the importance of various relationships in a knowledge graph through an attention mechanism, while KGCN (Wang, Zhao, Xie, Li and Guo, 2019) combines graph convolutional networks with knowledge graphs to extract relational features more effectively. KGCF (Chen et al., 2024) incorporates the user's social features into a recommendation framework. The user's interaction information is auto-encoded linearly by incorporating the social relationship graph into the user's interaction information and constructing a multi-layer graph filter. These models enhance the recommendation accuracy by incorporating external knowledge but often focus on deep diffusion scenarios. KGCE (Huang et al., 2024) proposes dynamic confidence weights to enhance the influence of entities across different attributes during information diffusion. KGIN (Wang et al., 2021) emphasizes the modelling user of intent and dynamic preferences rather than relying solely on static knowledge graph structures, aiming for more accurate preference predictions. However, these models may not be suitable for broad diffusion with short chains or weak temporal correlations. They do not explicitly capture the temporal dynamics of user preferences. User interests change over time, and static models may not adapt well to these changes.

Self-supervised learning, represented by contrastive learning, has received extensive attention. KGCL (Yang, Huang, Xia, & Li, 2022) applies random graph augmentation to mitigate noise and address long-tail issues, and KGIC (Zou et al., 2022) utilizes multi-level contrastive learning to capture information hierarchies within the knowledge graph. Similarly, deep learning and contrastive learning models, though they improve flexibility and robustness, often lack interpretability and are predominantly designed for deep diffusion rather than broad diffusion scenarios. While these models focus mainly on personalizing recommendations, our approach focus on the sharing recommendation task within broad diffusion contexts, aiming to optimize community-wide information sharing. This facilitates greater exposure and interaction with shared content, thereby expanding the reach of information across broader user groups.

3. Task formulation

In this section, we present relative definitions and task formulations before introducing model details.

Definition 1. *Information Diffusion.* User set U and item set V are main participants of information diffusion. Each diffusion step involves an inviter u_i^{inv} sharing an item v_j^{share}, and a voter u_k^{vot} clicking on the sharing link. If an item v_j^{share} can be forwarded by users $c(v_j^{\text{share}}) = \{u_1^{\text{vot}} \to u_2^{\text{vot}} \to \cdots \to u_k^{\text{vot}}\}$ in chronological order, the collection of diffusion chains $C = \{c(v_j^{\text{share}}) | j \in [V]\}$ exhibits the scenario of <u>deep</u> information diffusion. If each voter is independently influenced by $(u_i^{\text{inv}}, v_j^{\text{share}})$ in a set of diffusion steps $Quad(u_i^{\text{inv}}, v_j^{\text{share}}) = \{(u_1^{\text{vot}}, t_1^{\text{vot}}), (u_2^{\text{vot}}, t_2^{\text{vot}}), \ldots, (u_k^{\text{inv}}, t_k^{\text{vot}})\}$, the quad collection exhibits the scenario of <u>broad</u> information diffusion.

Definition 2. *Knowledge Graph.* Knowledge graph $\mathcal{G} = \{\mathcal{E}, \mathcal{R}\}$ is a directed heterogeneous graph composed of the entity set $\mathcal{E}$ and the relation set $\mathcal{R}$. If the $\mathcal{E}$ mainly consists of users U and user attributes, and the $\mathcal{R}$ contains social connections, the knowledge base manifests a <u>s</u>ocial <u>k</u>nowledge <u>g</u>raph (SKG). Similarly, if the $\mathcal{E}$ mainly consists of items V and item attributes, the knowledge base manifests an item knowledge graph. Specially, when extra user entities and user-item interactions are added, the item knowledge graph is extended to a <u>c</u>ollaborative <u>k</u>nowledge <u>g</u>raph (CKG).

Definition 3. *Classical Knowledge-driven Recommendation Task.* Given a knowledge graph $\mathcal{G} = \{\mathcal{E}, \mathcal{R}, \mathcal{T}\}$ and user-item interactions $\mathcal{I} = \{(u_i, y_{ij}, v_j) | u_i \in U, v_j \in V\}$, we adopt implicit feedback, and $y_{ij} = 1$ if u_i interacted with v_j. The objective is to obtain a model $\mathcal{F}(u, v | \mathcal{G}, \mathcal{I}, \Theta)$ that scores how well an item matches a user's interest. Θ denotes model parameters.

Definition 4. *Voter-oriented Prediction Task.* In the scenario of BID, the training dataset $\mathcal{D}_{\text{info}}$ is composed of diffusion steps represented as $Quad_i = (u_i^{\text{inv}}, v_j^{\text{share}}, u_k^{\text{vot}}, t_k^{\text{vot}})$. Each user has two roles of 'inviter' and 'voter', and t_k^{vot} is the timestamp when the voter u_k^{vot} clicked on the sharing link v_j^{share}. At the prediction stage, the objective is equivalent to complement quads of $(u_i^{\text{inv}}, v_j^{\text{share}}, t_j^{\text{share}}, ?)$, predict the u^{vot}, where t_j^{share} is the timestamp when the inviter u_i^{inv} shared the item v_j^{share}.

4. Matching phase

To simultaneously model both social relationships and personalized preferences, we employ the "Match & Rank" approach. The matching phase simplifies the problem into two separate knowledge-driven recommendation tasks, each focusing on one factor. Figs. 4 and 5 depict model details, employing similar logic of knowledge embedding and graph representation learning.

4.1. Retrieving voters based on social relations

The intuition underlying this retrieval path is that friends with close social interactions are more likely to become voters for future information diffusion. Therefore, triplets of SKG are constructed from '$inviter \xrightarrow{share} item$', '$voter \xrightarrow{click} item$' and '$user \xrightarrow{attr} value$'. We additionally introduce '$inviter \xrightarrow{social} voter$' triplets if a voter clicks on an item

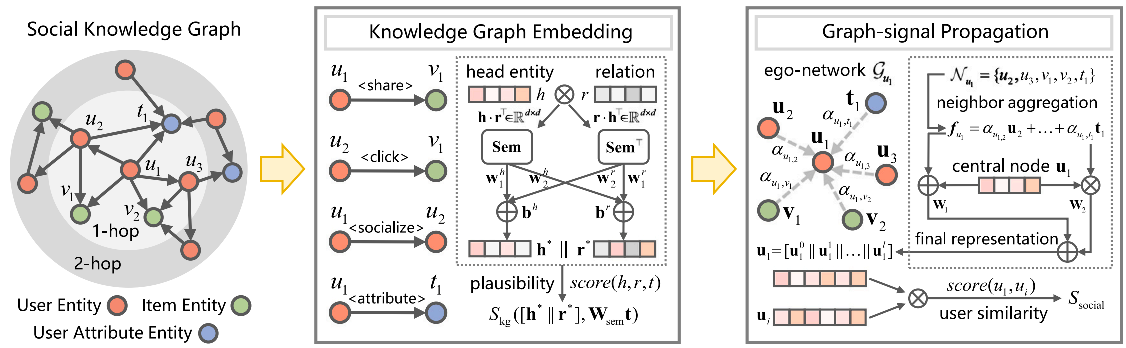


Fig. 4. The model architecture of similar user recommendation based on SKG. The knowledge base mainly reflects social relationships between users. We embed knowledge triplets through a semantic cross unit and utilize a graph attention network to propagate graph signals. Both triplet plausibility S_{kg} and user similarity S_{social} are required fitting ground truths and optimizing the model in a joint-learning scheme.

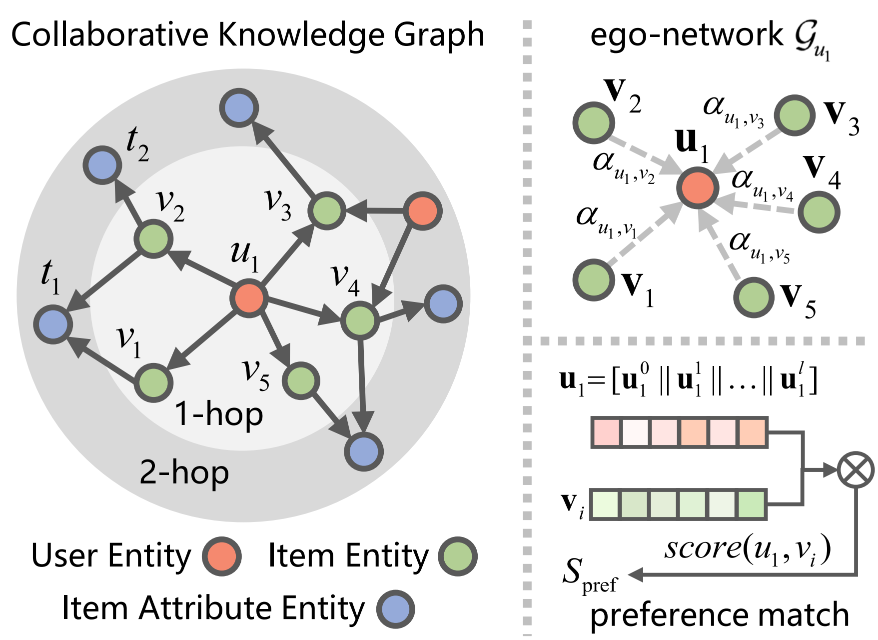


Fig. 5. Item audience recommendation based on CKG.

link shared by another inviter. As illustrated in Fig. 4, the primary objective of this module is to embed the triplets of the SKG into a high-dimensional vector space. The SKG primarily reflects the social relationships between users, and the embedding process is designed to effectively capture these relationships. Subsequently, we detail the retrieval process through the similar user recommendation model.

4.1.1. Module of knowledge graph embedding

Classical knowledge graph embedding algorithms (Bordes, Usunier, Garcia-Duran, Weston, & Yakhnenko, 2013; Lin, Liu, Sun, Liu, & Zhu, 2015; Wang, Zhang, Feng, & Chen, 2014) lack deep semantic crossing between entities and relations. Inspired by the knowledge-fusion network in EMKR (Gao et al., 2023), we devise a semantic crossing unit (SCU) to achieve the relation-aware knowledge embedding. Given a triplet (h, r, t), SCU first crosses each dimension of head entity and relation embeddings $\mathbf{h}, \mathbf{r} \in \mathbb{R}^d$:

$$\mathbf{Sem} = \mathbf{h} \cdot \mathbf{r}^{\top} = \begin{bmatrix} h^{(1)}r^{(1)} & \dots & h^{(1)}r^{(d)} \\ \dots & & \dots \\ h^{(d)}r^{(1)} & \dots & h^{(d)}r^{(d)} \end{bmatrix} \tag{1}$$

where, $h^{(d)}$ refers to the value of dth dimension in $\mathbf{h}$, and each term in the crossing matrix $\mathbf{Sem} \in \mathbb{R}^{d\times d}$ conforms to $h^{(i)}r^{(j)}, \forall (i,j) \in \{1,\dots,d\}^2$. Similarly, in the opposite direction, $\mathbf{Sem}^{\top} = \mathbf{r} \cdot \mathbf{h}^{\top}$.

SCU transforms the two crossing matrices back to entity and relation representations through vectors $\mathbf{w}_1^h, \mathbf{w}_2^h, \mathbf{w}_1^r, \mathbf{w}_2^r, \mathbf{b}^h$ and $\mathbf{b}^r$:

$$\begin{aligned} \mathbf{h}^* &= \mathbf{Sem} \cdot \mathbf{w}_1^h + \mathbf{Sem}^{\top} \cdot \mathbf{w}_2^r + \mathbf{b}^h \\ \mathbf{r}^* &= \mathbf{Sem} \cdot \mathbf{w}_2^h + \mathbf{Sem}^{\top} \cdot \mathbf{w}_1^r + \mathbf{b}^r \end{aligned} \tag{2}$$

The theoretical proof demonstrates that the semantic fusion process of SCU is analogous to the second-order feature crossing of Factorization Machine (FM). Specifically manifested in the L1 norm (taking the SCU calculation result $\mathbf{h}^*$ as an example):

$$Logit_{FM} = w_0 + \sum_{i=1}^{d} w_i x_i + \sum_{i=1}^{d} \sum_{j=i+1}^{d} \langle \mathbf{a}_i, \mathbf{a}_j \rangle x_i x_j \tag{3}$$

$$\begin{aligned} \|\mathbf{h}^*\|_1 &= \| \mathbf{h}\mathbf{r}^{\top}\mathbf{w}_1^h + \mathbf{r}\mathbf{h}^{\top}\mathbf{w}_2^r + \mathbf{b}^h \|_1 \\ &= \left| \sum_{i=1}^{d} \mid b^{h,(i)} \mid + \sum_{i=1}^{d} \sum_{j=1}^{d} \langle w_1^{h,(i)}, w_2^{r,(j)} \rangle h^{(i)} r^{(j)} \right| \end{aligned} \tag{4}$$

The formula (3) represents the prediction function of FM, where the initial term $w_0 + \sum_{i=1}^{d} w_i x_i$ aligns with the basic definition of linear regression. The subsequent term is accounts for the second-order interactions between features, with $\mathbf{a}_i$ represents the weight vector of the ith feature. The main difference between formulas (3) and (4) lies in the computation of the $\langle \cdot, \cdot \rangle$; the former calculates the dot product of weight vectors, while the latter is the sum of two scalars. Consequently, compared to FM, SCU has fewer parameters and higher training efficiency. Moreover, in comparison with relation-specific matrices applied in Lin et al. (2015) and Wang et al. (2014), $\mathbf{w}^{h;r}_{1;2}$ and $\mathbf{b}^{h;r}$ accomplish more generalized knowledge sharing throughout the graph structure.

We concatenate the outputs $(\mathbf{h}^*, \mathbf{r}^*)$ of SCU and compute the triplet plausibility with tail entity embedding $\mathbf{t} \in \mathbb{R}^d$:

$$score(h,r,t) = S_{\mathrm{kg}}(\mathbf{W}_{\mathrm{sem}}[\mathbf{h}^* \parallel \mathbf{r}^*], \mathbf{t}) \tag{5}$$

where, $\mathbf{W}_{\mathrm{sem}} \in \mathbb{R}^{d\times 2d}$ is a parametric weight matrix that transforms the concatenated embedding $\mathbf{h}^* \parallel \mathbf{r}^*$ to capture the semantic interactions between the head entity and the relation, and the inner product is used to calculate S_{kg}, which computes the plausibility of the triplet by measuring the compatibility between the transformed combined embedding and the tail entity embedding $\mathbf{t}$. Finally, we utilize BPR (Rendle, Freudenthaler, Gantner, & Schmidt-Thieme, 2009) to encourage higher scores for positive triplets:

$$\mathcal{L}_{\mathrm{kg}} = \sum_{(h,r,t)\in\mathcal{G}} -\ln\sigma\left(score(h,r,t) - score(h,r,t')\right) \tag{6}$$

where, $\sigma(\cdot)$ is the Sigmoid function, and $(h,r,t') \notin \mathcal{G}$ is the negative triplet that replaces the tail entity t in (h,r,t).

4.1.2. Module of graph-signal propagation

When aggregating neighbour information, we neglect the directionality of relations, instead weighing the importance of different neighbours through attention scores $\alpha(h,r,t)$.

To reduce parameters' size and enhance knowledge sharing across various triplets, we define the attention score as:

$$\alpha(h,r,t) = \mathrm{LeakyReLU}(\mathbf{t}^\top \mathbf{W}_{\mathrm{sem}}[\mathbf{h} \parallel \mathbf{r}]) \tag{7}$$

$$\alpha(h,r,t) = \frac{\exp(\alpha(h,r,t))}{\sum_{(h,r',t')\in\mathcal{G}_h}\exp(\alpha(h,r',t'))} \tag{8}$$

where, $\mathcal{G}_h$ is an ego-network with h as the central node. Weighing the degree of importance of different neighbours, this allows the model to focus more on relevant neighbours, even in sparse regions of the graph. We reuse the weight matrix $\mathbf{W}_{\mathrm{sem}}$ from the knowledge embedding module to incorporate semantic information into attention scores.

At the lth graph-signal propagation layer, the aggregated result is the attentive sum of:

$$\boldsymbol{f}^l_h = \sum_{(h,r,t)\in\mathcal{G}_h} \alpha(h,r,t)\cdot \mathbf{t}^{l-1} \tag{9}$$

To further address sparsity, the Graph Attention Network propagates information through multiple layers, aggregating information from multi-hop neighbours. This helps in capturing higher-order relationships and mitigates the impact of sparse local neighbourhoods. Even if an inactive user has few direct connections, the model can still leverage information from more distant neighbours through multi-hop propagation. Then, we follow the practice of bi-interaction aggregator (Wang, He et al., 2019) to represent the lth layer's central node:

$$\mathbf{h}^l = \mathrm{MLP}_1\left(\mathbf{h}^{l-1} + \boldsymbol{f}^l_h\right) + \mathrm{MLP}_2\left(\mathbf{h}^{l-1} \odot \boldsymbol{f}^l_h\right) \tag{10}$$

where, $\odot$ denotes element-wise product, and $\mathrm{MLP}(\cdot)$ contains a fully connected layer and an activation layer of LeakyReLU.

Finally, given two user entities (u_i, u_j), we concatenate each layer's output to compute their similarity:

$$score(u_i,u_j) = S_{\mathrm{social}}\left([\mathbf{u}^0_i \| \cdots \| \mathbf{u}^l_i], [\mathbf{u}^0_j \| \cdots \| \mathbf{u}^l_j]\right) \tag{11}$$

where, $\mathbf{u}_i$ is the user embedding of u_i which represent the user u_i at different layers of the graph signal propagation process. Each layer's embedding captures different aspects of the user's social relationships and preferences. The concatenation of these embeddings $[\mathbf{u}^0_i \| \cdots \| \mathbf{u}^l_i]$ provides a comprehensive representation of the user u_i. We adopt inner product to calculate S_{social} which measures the cosine similarity between the two user embeddings. Finally, we utilize BPR loss to optimize this module:

$$\mathcal{L}_{\mathrm{social}} = \sum\nolimits_{\mathcal{D}_{\mathrm{info}}} -\ln\sigma(score(u^{\mathrm{inv}}_i, u^{\mathrm{vot}}_k) - score(u^{\mathrm{inv}}_i, u^{\neg\mathrm{vot}}_k)) \tag{12}$$

In terms of a diffusion step $Quad_i = (u^{\mathrm{inv}}_i, v^{\mathrm{share}}_j, u^{\mathrm{vot}}_k, t^{\mathrm{vot}}_k)$, $Quad_i \in \mathcal{D}_{\mathrm{info}}$, the loss function intends to increase the similarity of the positive sample $(u^{\mathrm{inv}}_i, u^{\mathrm{vot}}_k)$ and decrease the similarity of the negative sample $(u^{\mathrm{inv}}_i, u^{\neg\mathrm{vot}}_k) \notin \mathcal{D}_{\mathrm{info}}$. The whole similar user recommendation model is trained in a joint-learning scheme, *i.e.*, $\mathcal{L} = \mathcal{L}_{\mathrm{kg}} + \mathcal{L}_{\mathrm{social}}$.

4.2. Retrieving voters based on user preference

We construct knowledge graphs by incorporating additional user and item attributes, which enriches the topological connections and provides more context for inactive users. This augmentation helps in reducing the impact of sparsity by providing more informative features for embedding. The underlying intuition of this retrieval path is that users primarily share or click items based on their personalized preference. Therefore, triplets of CKG are constructed from '$user \xrightarrow{interact} item$' and '$item \xrightarrow{attr} value$'. Due to the identical setting of graph attention network and knowledge graph embedding, Fig. 5 only illustrates the main differences in graph structure and prediction objective.

Given a user entity u_i and an item entity v_j, their matching degree is computed by inner product of:

$$score(u_i,v_j) = S_{\mathrm{pref}}\left([\mathbf{u}^0_i \| \cdots \| \mathbf{u}^l_i], [\mathbf{v}^0_j \| \cdots \| \mathbf{v}^l_j]\right) \tag{13}$$

The training loss is also acquired by BPR:

$$\mathcal{L}_{\mathrm{pref}} = \sum\nolimits_{(u_i,v_j)\in\mathcal{I}} -\ln\sigma(score(u_i,v_j) - score(u_i,v'_j)) \tag{14}$$

where, $\mathcal{I}$ denotes user-item interactions converted from information diffusion dataset $\mathcal{D}_{\mathrm{info}}$, and $(u_i, v'_j) \notin \mathcal{I}$ is a negative sample replacing the real interacted item v_j. The whole item audience recommendation model is trained end-to-end by $\mathcal{L} = \mathcal{L}_{\mathrm{kg}} + \mathcal{L}_{\mathrm{pref}}$.

4.3. Retrieval path merging

We can acquire two sets of retrieved voters from: (1) The recommendation model in Section 4.1 retrieves similar users $U^{\mathrm{social}}_{u_i}$ in terms of the inviter u^{inv}_i; (2) The recommendation model in Section 4.2 retrieves item audience $U^{\mathrm{pref}}_{v_j}$ in terms of the shared item v^{share}_j. The two factors of social relationships and matching preference are respectively covered in two retrieval paths, of which each measures user-user or user-item scores by own scoring function $score(\cdot,\cdot)$. The final candidate users are the union set of two top-K lists, *i.e.*, $U^{\mathrm{cand}}_{u_i,v_j} = U^{\mathrm{social}}_{u_i} \cup U^{\mathrm{pref}}_{v_j}$, $|U^{\mathrm{cand}}_{u_i,v_j}| = 2K$.

5. Ranking phase

The overall framework of K-BID is depicted in Fig. 6. Above knowledge retrieval models serve as the foundation for the downstream temporal prediction model. Apart from providing retrieved voters $U^{\mathrm{cand}}_{u_i,v_j}$ to narrow the solution domain, the matching phase also transfers social and preference knowledge to the embedding tables in the ranking phase. Note that the temporal prediction model is no longer limited to either social or preference factors. Instead, it assesses actual intent of every voter in a more comprehensive manner, based on historical behaviours, shared items and both user parties.

5.1. Knowledge transferring

Since behaviour patterns vary with different user roles, we separately maintain embedding tables for inviters $\mathbf{U}^{\mathrm{inv}}$ and voters $\mathbf{U}^{\mathrm{vot}}$. Meanwhile, the SKG-based retrieval model $\mathcal{F}_{\mathrm{SKG}}$ learns the similarity among users and owns pretrained weights $\mathbf{U}^{\mathrm{social}}$ encompassing abundant social information. The CKG-based retrieval model $\mathcal{F}_{\mathrm{CKG}}$ concentrates on user-item matching degrees and owns pretrained weights

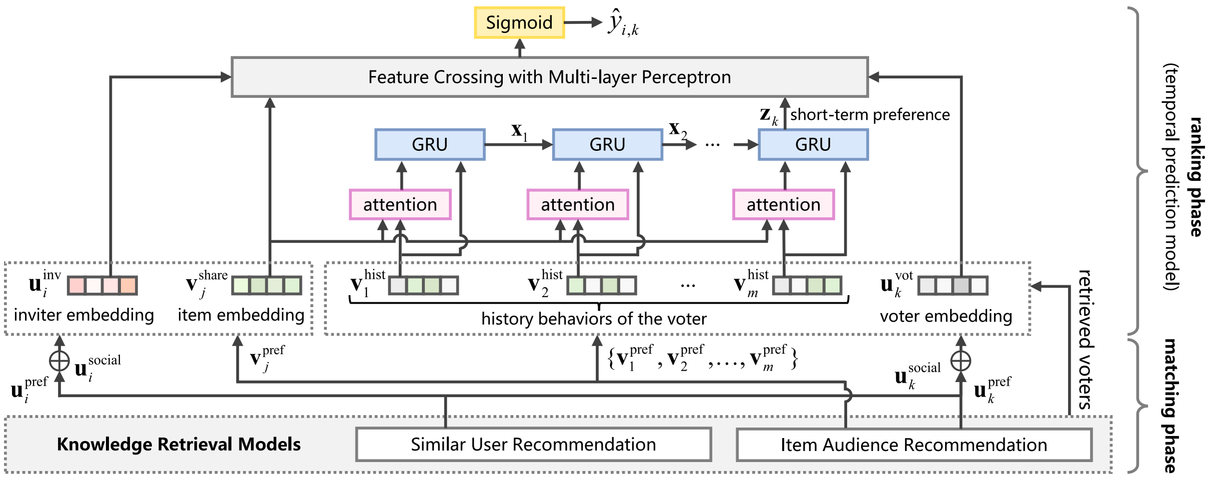


Fig. 6. The overall framework of our proposed knowledge-driven broad information diffusion model (K-BID). The matching phase provides retrieved voters and pretrained embeddings. The temporal refinement process further extracts voters' evolving preferences and weighs the importance of social and preference factors for each diffusion step.

$\{\mathbf{U}^{\text{pref}}, \mathbf{V}^{\text{pref}}\}$ encompassing preference information. Hence, we design the knowledge transferring as:

$$\begin{cases} \mathbf{U}^{\text{inv}} := \lambda_1 \mathbf{U}^{\text{social}} + \lambda_2 \mathbf{U}^{\text{pref}} \\ \mathbf{V} := \mathbf{V}^{\text{pref}} \\ \mathbf{U}^{\text{vot}} := \lambda_1 \mathbf{U}^{\text{social}} + \lambda_2 \mathbf{U}^{\text{pref}} \end{cases}, \begin{matrix} \mathbf{U}^{\text{social}} \in \mathcal{F}_{\text{SKG}} \\ \mathbf{U}^{\text{pref}}, \mathbf{V}^{\text{pref}} \in \mathcal{F}_{\text{CKG}} \end{matrix} \tag{15}$$

where, $(\lambda_1 \mathbf{U}^{\text{social}} + \lambda_2 \mathbf{U}^{\text{pref}})$ means that voters may exhibit both social relationships with inviters and personalized preference for shared items. The transferring scheme promotes the temporal prediction model to simultaneously consider these factors. The coefficients λ_1, λ_2 are determined by respective contributions of two knowledge retrieval models to the voter-oriented prediction task.

5.2. Short-term preference representation

User interests are evolving over time, and recently interacted items are more indicative of future voting behaviours. Inspired by the news recommendation model in MFF (Huang et al., 2023), we utilize shared items and voters' history behaviours to capture the preference evolution of voters. For one training sample $Quad_i = (u_i^{\text{inv}}, v_j^{\text{share}}, u_k^{\text{vot}}, t_k^{\text{vot}})$, we obtain M items interacted with u_k^{vot} (regardless of sharing or click behaviours) prior to timestamp t_k^{vot} as the voting history $V_{u_k}^{<t_k} = \{v_m^{\text{hist}} | m \in [1, M]\}$. Since GRU reports higher efficiency than LSTM (Mao & Sejdić, 2023), we model short-term preference through the reset gate $\mathbf{G}^r$ and update gate $\mathbf{G}^u$:

$$\mathbf{x}_m = \text{GRU}\left(\mathbf{v}_m^{\text{hist}}, \mathbf{x}_{m-1}, \mathbf{G}^r, \mathbf{G}^u\right), \mathbf{v}_m^{\text{hist}} \in V_{u_k}^{<t_k} \tag{16}$$

where, $\mathbf{x}_{m-1}$ is the $(m-1)$th hidden state, and we regard the final hidden state $\mathbf{x}_m$ as the encoded preference $\mathbf{z}_k$ of the voter u_k^{vot}. If following the practice of MFF, $\mathbf{z}_k$ will be further crossed with v_j^{share} to evaluate the matching degree of u_k^{vot}'s preference. However, the highly condensed feature $\mathbf{z}_k$ has lost fine-grained history information. Overlapping preference patterns should be captured at an earlier stage.

We opt for the attentive update gate (Zhou et al., 2019) to diminish the impact of irrelevant history behaviours on $\mathbf{z}_k$:

$$\alpha\left(v_j^{\text{share}}, v_m^{\text{hist}}\right) = \frac{\exp\left(\mathbf{v}_j^{\text{share}} \mathbf{W}_{\text{att}} \mathbf{v}_m^{\text{hist}}\right)}{\sum_{q=1}^{M} \exp\left(\mathbf{v}_j^{\text{share}} \mathbf{W}_{\text{att}} \mathbf{v}_q^{\text{hist}}\right)} \tag{17}$$

$$\mathbf{G}^u = \alpha\left(v_j^{\text{share}}, v_m^{\text{hist}}\right) \mathbf{G}^u \tag{18}$$

where, $\mathbf{W}_{\text{att}}$ is a parametric weight matrix. The higher the similarity between the shared item v_j^{share} and u_k^{vot}'s interacted item v_m^{hist}, the higher the attention score $\alpha(\cdot,\cdot)$ is, and the more information of the current time step and the update gate $\mathbf{G}^u$ can be retained.

5.3. Voter prediction with knowledge retrieval

After the knowledge transferring and preference representation, we have held social knowledge and long-short term preference. So, which factor will play a key role and serve as a 'trigger' of information diffusion?

We feed $[\mathbf{u}_i^{\text{inv}} \parallel \mathbf{v}_j^{\text{share}} \parallel \mathbf{u}_k^{\text{vot}} \parallel \mathbf{z}_k]$ into a two-layer MLP and a Sigmoid function, to get the predicted score:

$$\hat{y}_{i,k} = \sigma\left(\text{MLP}\left([\mathbf{u}_i^{\text{inv}} \parallel \mathbf{v}_j^{\text{share}} \parallel \mathbf{u}_k^{\text{vot}} \parallel \mathbf{z}_k]\right)\right) \tag{19}$$

which means the diffusion probability from the inviter u_i^{inv} to the voter u_k^{vot}.

Finally, we measure the Binary Cross Entropy to optimize the temporal prediction model:

$$\mathcal{L}_{\text{tar}} = \sum_{\mathcal{D}_{\text{info}}} -\log(\hat{y}_{i,k}) - N_{\text{neg}} \mathbb{E}_{u_{k'}^{\text{vot}} \sim \mathcal{P}_{\text{neg}}(u_i^{\text{inv}})} \log(1 - \hat{y}_{i,k'}) \tag{20}$$

where, N_{neg} is the number of negative samples. We choose users $u_{k'}^{\text{vot}}$ who have not interacted with the inviter u_i^{inv} from the negative sampling distribution $\mathcal{P}_{\text{neg}}(u_i^{\text{inv}})$. At the model test stage, for a test sample $(u_i^{\text{inv}}, v_j^{\text{share}}, t_j^{\text{share}}, u_k^{\text{label}})$ with u_k^{label} as the ground truth voter, we obtain $U_{u_i,v_j}^{\text{cand}}$ including:

- Top-K similar users $U_{u_i}^{\text{social}}$ through the SKG-based retrieval model $\mathcal{F}_{\text{SKG}}$ and the inviter u_i^{inv};
- Top-K item audience $U_{v_j}^{\text{pref}}$ through the CKG-based retrieval model $\mathcal{F}_{\text{CKG}}$ and the shared item v_j^{share}.

After the temporal refinement, it is expected for the ground truth voter u_k^{label} to be ranked higher in the sorted candidate list $U_{u_i,v_j}^{\text{cand}} = U_{u_i}^{\text{social}} \cup U_{v_j}^{\text{pref}}$. The complete approach comes to an end.

5.4. Time complexity analysis

The Similar User Recommendation (SUR) model and the Item Audience Recommendation (IAR) model are two key components of our proposed K-BID framework, each designed to address different aspects of the recommendation task within the matching phase. Their time complexity can be estimated by several key steps: (1) The SCU module performs operations on each dimension of entity and relation embeddings. The crossing matrix **Sem** requires a time complexity of $O(d^2)$, where d is the embedding dimension. Subsequent operations, such as the reconstructed head embedding $\mathbf{h}^*$ and the triplet plausibility function score(h,r,t), also maintain the $O(d^2)$ complexity. If the number of triplets is $|\mathcal{T}|$, the time complexity of the SCU module is $O(|\mathcal{T}| \cdot d^2)$.

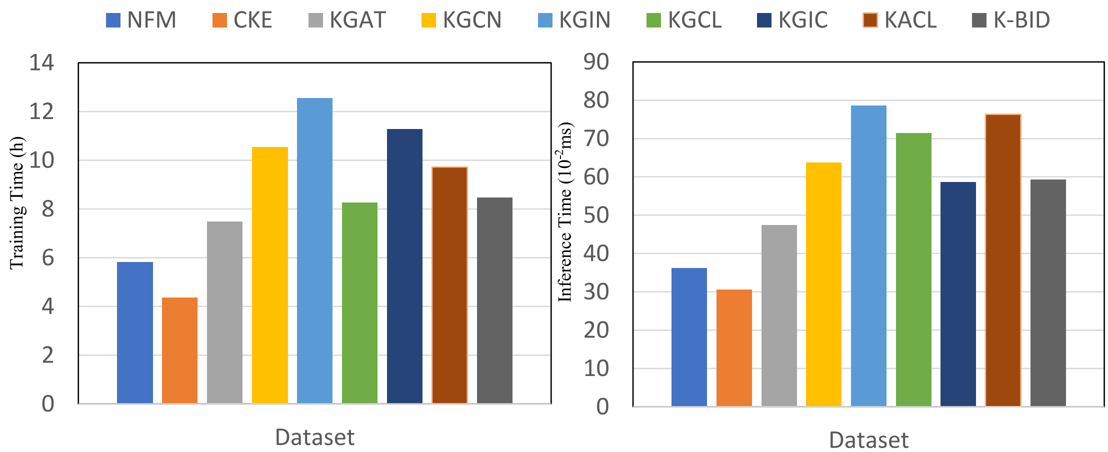


Fig. 7. Training time and inference time on the dataset.

(2) The graph attention mechanism aggregates information from neighbours of a user node. The main computational overhead is the attention score calculation $\alpha(h,r,t)$, which contributes to $O(d^2)$ complexity. By reusing the weight matrix $\mathbf{W}_{\mathbf{sem}}$ from the SCU module, we reduce the parameter size and enhance knowledge sharing across various triplets, improving training efficiency. If each node has an average degree of k and there are $|\mathcal{E}|$ entities in the graph, the time complexity for L layers of graph attention propagation is $O(k \cdot |\mathcal{E}| \cdot L \cdot d^2)$. In summary, considering that the differences between SUR and IAR models only exist in the construction of knowledge graphs and loss functions, both models have an overall time complexity of $O((|T| + kL|\mathcal{E}|) \cdot d^2)$. Another noteworthy point is that in the real industrial deployment, the SUR and IAR models run in parallel as two separate retrieval paths in the matching phase. Then, we analyse the next temporal prediction model in the ranking phase: (1) The knowledge transferring adds pre-trained embeddings to the embedding tables. This has a time complexity of $O(U \cdot d)$, where U is the number of users. (2) The short-term preference representation involves an attentive update gate apart from conventional GRU operations. The additional gate requires an operation $\alpha(v_j^{\text{share}}, v_m^{\text{hist}})$ between the shared item embedding and the history item embedding for each item in the history. If N_{hist} is the number of history items, its overall complexity is $O(N_{hist}^2 \cdot d^2)$. Therefore, the time complexity of K-BID is approximated as a $O((|T| + kL|\mathcal{E}|) \cdot d^2) + \mathrm{O}(U \cdot d + N_{\text{hist}}^2 \cdot d^2)$.

To analyse the computational efficiency of K-BID compared to the baselines on dataset, we provide the training time of the recommendation models on the training set and the inference time on the test set in Fig. 7. We calculate the average inference time of a single user based on the overall inference time of the test set. This average inference time per user is used as the inference time metric for dataset in Fig. 7. K-BID's training time is slightly higher than some baselines (e.g., KGAT, KGCL), K-BID achieves a balance between efficiency and performance through its two-phase "Match & Rank" architecture. Specifically, the parallel design in the matching phase (independent retrieval of social and collaborative graphs) reduces computational redundancy. Although K-BID may not have the lowest runtime, its balance between running time and performance makes it a robust choice for practical applications.

6. Experiments

6.1. Datasets

We perform experiments on a real-life E-commerce dataset sourced from a Tianchi Competition (https://tianchi.aliyun.com/dataset/162145), including user profiles, item attributes and sharing records based on '(*inviter*, *shared item*, *voter*, *timestamp*)'. To the best of our knowledge, we are the first work to explore the broad information diffusion scenario on the dataset. The training set comprises over 600,000 samples spanning 306 days from December 27, 2021 to October 29, 2022. The validation set comprises over 110,000 samples spanning 31 days from October 29, 2022 to November 29, 2022. The test set comprises over 110,000 samples spanning 28 days from November 29, 2022 to December 27, 2022. There is an overlap in user groups but not in time periods, ensuring that the social and preference patterns in the training set are consistently reflected in the test set. We normalized the timestamps to a consistent format and converted them into a uniform time scale (e.g., seconds since a fixed start time) to facilitate temporal analysis. In addition, we considered noise processing. For example, a case in which a person clicked on a large number of items (hundreds or thousands of clicks) in a short period (e.g., an hour) is not practical. Furthermore, we ensured the consistency and correct mapping of all user and item IDs across the dataset to maintain data integrity. The ratio of the training, validation, and test sets is 8:1:1. Although we only used this one dataset for evaluation, we believe that sufficient evaluation results can demonstrate the feasibility of our model in the broad diffusion scenario.

We examine the lengths of the information diffusion chains in the training set to verify whether the same item information can be forwarded by multiple users. Fig. 8 shows the statistical results with other popular information diffusion datasets (LastFM Celma Herrada, 2010, Irvine Opsahl & Panzarasa, 2009 and Memetracker Bourigault, Lamprier, & Gallinari, 2016) released in Yang et al. (2021). In our sharing records, the number of chains with length 1 accounts for 98.94% of the total chains and 99.43% of the total training samples. However, taking LastFM as an example, chain lengths exceeding 100 are quite prevalent. This observation shows that previous models based on deep information diffusion are not suitable for broad information diffusion scenarios. It also explains the challenge and motivation of discussing information diffusion in this paper.

Table 1 summarizes several statistics of the BID datasets. User profiles include 'gender', 'age' and 'platform level' and item attributes contain 'category-1', 'category-2', 'brand' and 'shop'. Table 2 summarizes the statistics of social and collaborative knowledge graphs, both of which are constructed from the BID datasets and reach the scale of millions.

In the item audience recommendation task, we apply the 5-core setting to ensure at least five interaction items (regardless of sharing or click behaviours) for each user. In the similar user recommendation task, because some inviters connect solely with fixed voters, we do not limit the minimum number of social interactions between users. In the ranking phase, we exclude diffusion steps $\{u_i^{\text{inv}}, v_j^{\text{share}}, u_k^{\text{vot}}, t_k^{\text{vot}}\}$ whose voter u_k^{vot} has not interacted with any items prior to timestamp t_k^{vot}.

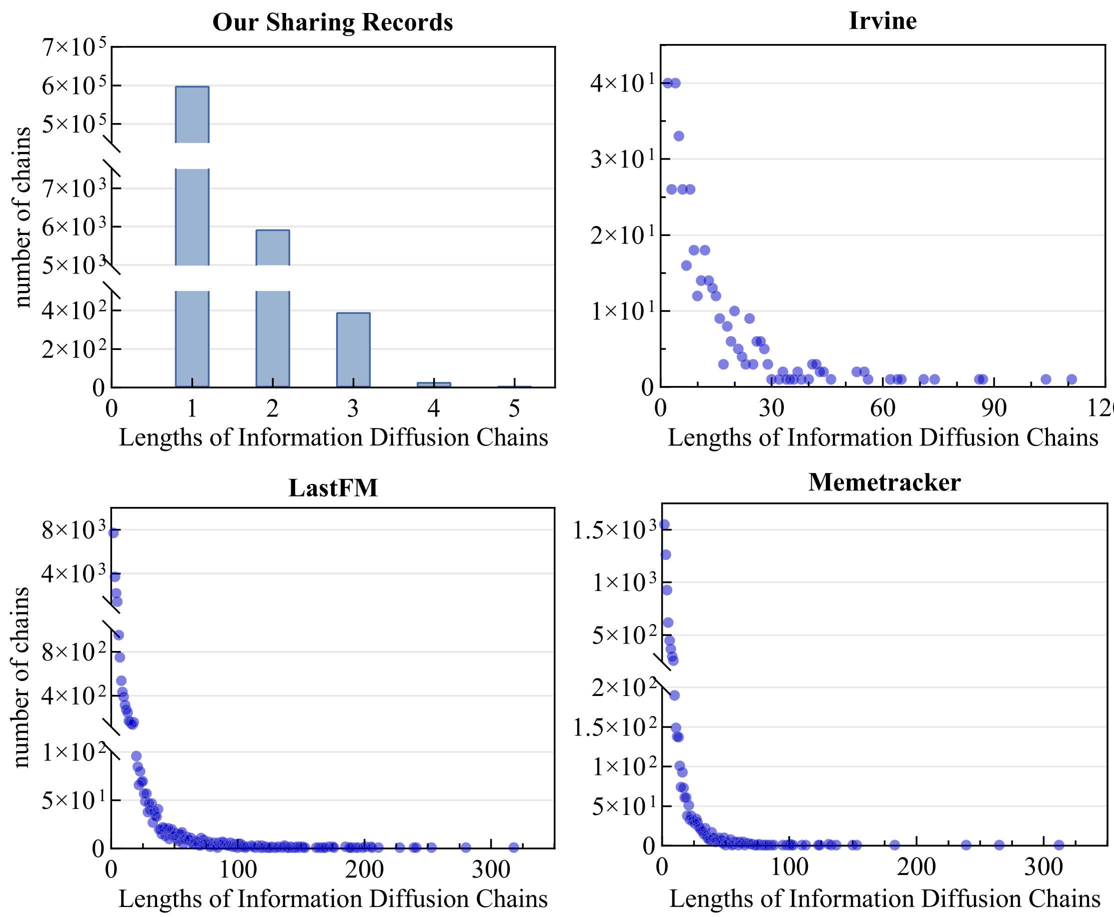


Fig. 8. Comparison between lengths of information diffusion chains. For instance, if an item v_1 is shared by users '$u_1 \rightarrow u_2 \rightarrow u_3$', the chain length equals 3.

Table 1
Statistics of the broad information diffusion dataset.

Fields	Values	Fields	Values
Users	115,849	Training days	306
Items	438,164	Validation days	31
Training samples	602,679	Test days	28
Validation samples	119,527	Avg shares per day	1963
Test samples	118,424	Avg shares per user	30.87

Table 2
Statistics of knowledge graphs.

Knowledge graphs	Fields	Values
Social knowledge graph	Entities	554,035
	Entity types	5
	Relation types	6
	Triplets	1,476,546
Collaborative knowledge graph	Entities	699,916
	Entity types	6
	Relation types	5
	Triplets	1,752,656

6.2. Evaluation metrics

When the labelled voter is present in the prediction list and located in a more forward position, the model performance is better. Therefore, we select three typical metrics:

- Precision@K measures the proportion of labels present in a prediction list with length K.
- MRR@K measures the mean reciprocal ranking of the label's position in a prediction list with length K.
- AUC@K is the area under Precision-Recall curve and represents the probability that the positive sample can be ranked ahead of the negative sample.

K is set amongst $\{5, 10, 20\}$. Due to one test sample only involving one label in our sharing records, the value range of Precision@K is [0, $1/K$], and others are [0, 1].

6.3. Baselines

Past information diffusion-based models are deficient owing to the intrinsic limitation of short information diffusion chains in BID. We mainly perform experiments on representative knowledge-driven recommendation models:

- **NFM** (He & Chua, 2017) implements Factorization Machine by neural networks. We feed ids of an inviter, a shared item, a voter and their knowledge into the model.
- **CKE** (Zhang et al., 2016) embeds triplet knowledge by TransR and combines semantic information with collaborative filtering.
- **KGAT** (Wang, He et al., 2019) weighs the importance of neighbours in attentive propagation layers and embeds knowledge by TransR.
- **KGCN** (Wang, Zhao et al., 2019) exploits graph convolutional operations to capture multi-hop preference in receptive fields.
- **KGIN** (Wang et al., 2021) mines user intents from user-item interactions and perform graph representation learning with intents.
- **KGCL** (Yang et al., 2022) mitigates the knowledge noise and long-tail entity distribution through a contrastive learning strategy.

Table 3
Overall performance comparison.

Obj.	Model	Prec@K (%)			MRR@K (%)			AUC@K (%)		
		5	10	20	5	10	20	5	10	20
Soc.	NFM	5.78	3.91	2.30	17.15	18.84	19.60	18.74	29.26	39.64
	CKE	5.96	4.25	2.67	17.19	18.89	19.66	18.85	28.20	38.71
	KGAT	7.67	5.17	3.10	21.08	22.89	23.62	23.49	35.32	46.81
	KGCN	6.58	4.64	2.80	19.99	21.94	22.70	22.24	33.61	45.15
	KGIN	7.27	5.05	$\underline{3.14}$	22.03	23.42	23.83	24.40	35.05	45.29
	KGCL	6.71	4.13	2.20	21.84	23.21	23.62	24.34	35.02	45.18
	KGIC	7.79	$\underline{5.19}$	3.12	22.07	23.87	24.78	24.11	$\underline{36.13}$	$\underline{47.12}$
	KACL	$\underline{7.84}$	5.22*	3.16*	$\underline{22.40}$	$\underline{24.09}$	$\underline{24.86}$	$\underline{24.47}$	36.07	47.03
	SUR	7.92*	5.16	3.07	22.86*	24.49*	25.19*	25.48*	36.49*	47.16*
Pref.	NFM	1.27	0.88	0.60	3.80	4.13	4.35	4.14	5.98	8.29
	CKE	1.46	1.06	0.69	4.13	4.55	4.78	4.54	6.85	9.67
	KGAT	1.49	1.07	0.73	4.40	4.82	5.09	4.79	7.05	10.02
	KGCN	1.41	1.02	0.67	4.32	4.70	4.95	4.59	6.86	9.96
	KGIN	$\underline{1.83}$	1.36*	0.98*	5.30	5.88	$\underline{6.24}$	5.86	$\underline{8.72}$	$\underline{12.66}$
	KGCL	1.54	1.14	0.80	4.66	5.15	5.47	5.03	7.42	10.71
	KGIC	1.65	1.22	0.87	5.31	5.76	6.04	5.76	8.15	11.07
	KACL	1.76	1.20	0.79	$\underline{5.48}$	$\underline{5.90}$	6.16	$\underline{5.98}$	8.32	11.24
	IAR	1.86*	$\underline{1.35}$	$\underline{0.96}$	5.55*	6.09*	6.49*	6.04*	8.82*	12.78*
K-BID		**9.03**	**5.71**	**3.29**	**26.70**	**28.33**	**28.94**	**29.81**	**41.54**	**52.23**

The starred* and underlined values respectively denote the best and next best results in Soc. or Pref. The **bold** values are the overall optimal results.

- **KGIC** (Zou et al., 2022) utilizes layer-wise self-supervised signals to handle the unbalanced information in knowledge graphs.
- **KACL** (Wang et al., 2023) deals with the issues of interaction domination and task-irrelevant noise in knowledge graphs.
- **SUR** refers to our proposed similar user recommendation model in the matching phase.
- **IAR** refers to our proposed item audience recommendation model in the matching phase.

6.4. Experimental settings

For a fair comparison, we fix several basic hyper-parameters for all baselines and our K-BID: embedding dimensions (such as entity embeddings and relation embeddings) are 64; the batch size is 2048; the optimizer is Adam; the initializer is Xavier; the deep learning framework is PyTorch. We apply a grid search for hyper-parameters: the learning rate is tuned amongst $\{10^{-4}, 10^{-3}, 10^{-2}\}$, the coefficient of L2 normalization is searched in $\{10^{-5}, 10^{-4}, \ldots, 10^{-1}\}$, the temperature is tuned in $\{0.5, 0.6, 0.7, 0.8, 0.9\}$ and the number of GNN layers L is tuned in $\{1, 2, 3\}$ for GNN-based methods. In addition, we employ the node dropout technique for NFM and KGAT, where the ratio is tuned in $\{0.1, 0.2, 0.3\}$. For KGIN, the number of user intents is set as 4 and the number of relational path aggregation layers is set as 3 following its suggestion. For KGCL, the truncation probability is tuned in $\{0.6, 0.7, 0.8, 0.9\}$. For KGIC, the local and non-local triple set size are limited to 40 and 128 respectively. Other special hyper-parameters are decided by empirical studies in their own papers. We record the metrics at every 10 epochs and early stop a model training process if MRR@5 does not increase for 10 consecutive records. All of the experiments are conducted on a Windows platform using a single NVIDIA RTX 3090 GPU with 24 GB memory.

More settings of K-BID are as follows: (1) In the matching phase, we set 3 GNN layers for SUR and 1 GNN layers for IAR with degressive output dimensions $[64, 32, 16]$. The edge dropout rates are set to 0.1 for each GNN layer. To prevent overfitting, we append L2 regularization to both knowledge retrieval models. The regularization coefficient is 10^{-4}, and regularized parameters are tail entity embeddings $\mathbf{t}$ and encoded tensors $\{\mathbf{h}^*, \mathbf{r}^*\}$ in SCU. The number of retrieved voters $U_{u_i}^{\text{social}}(U_{v_j}^{\text{pref}})$ is 20. (2) In the ranking phase, we set $\lambda_1 = 0.8, \lambda_2 = 0.2$ based on the model performance of SUR and IAR. The hidden layers in MLP adopt output dimensions of $[128, 128]$ and dropout rates of 0.1. The maximum sequence length of voters' history behaviours $V_{u_k}^{<t_k}$ is 5.

6.5. Experimental results

6.5.1. Overall comparison

Owing to the semantic differences between social and collaborative knowledge graphs, we employ baselines to learn the user-user similarities and user-item matching degrees separately. The overall performance in two different modelling objectives is presented in Table 3, where the 'Soc.' refers to the social objective of (12), and the 'Pref.' refers to the preference objective of (14). Based on the experimental results, we have observations as follows:

- **Our proposed K-BID achieves the best performance.** Compared with all baselines, K-BID achieves the optimal performance in terms of Precision, MRR and AUC metrics with K amongst $\{5, 10, 20\}$. For the 'Soc.' objective, K-BID improves by 14.02%, 16.80% and 16.99% over the sub-optimal results in three metrics with $K = 5$. K-BID significantly outperforms in comparison to state-of-the-art baselines with $p < 0.05$ by adopting Wilcoxon signed rank statistical test (Shani & Gunawardana, 2011). This non-parametric test is particularly suitable for our dataset, which may not follow a normal distribution. We attribute the improved performance to: (1) Effective social and preference knowledge embedding in the matching phase; (2) Effective knowledge transferring scheme from the knowledge retrieval models to the temporal prediction model.
- **The social factor is more influential than the preference factor.** Model performance *w.r.t* the 'Soc.' objective is significantly higher than that *w.r.t* the 'Pref.' objective. Their best results respectively differ by 6.06, 17.31 and 19.44 in {Precision, MRR, AUC}@5 (%). Note that, it does not suggest the invalidity of the preference factor. Instead, both factors significantly influence on information diffusion.
- **The ranking phase is necessary for the performance improvement.** Benefiting from our semantic cross unit and graph attention network, a single knowledge retrieval model also has a competitive performance. However, we can observe that the improvements of SUR and IAR are not remarkable. The ranking phase is essential for refining the retrieval the lists and enhancing the performance further.

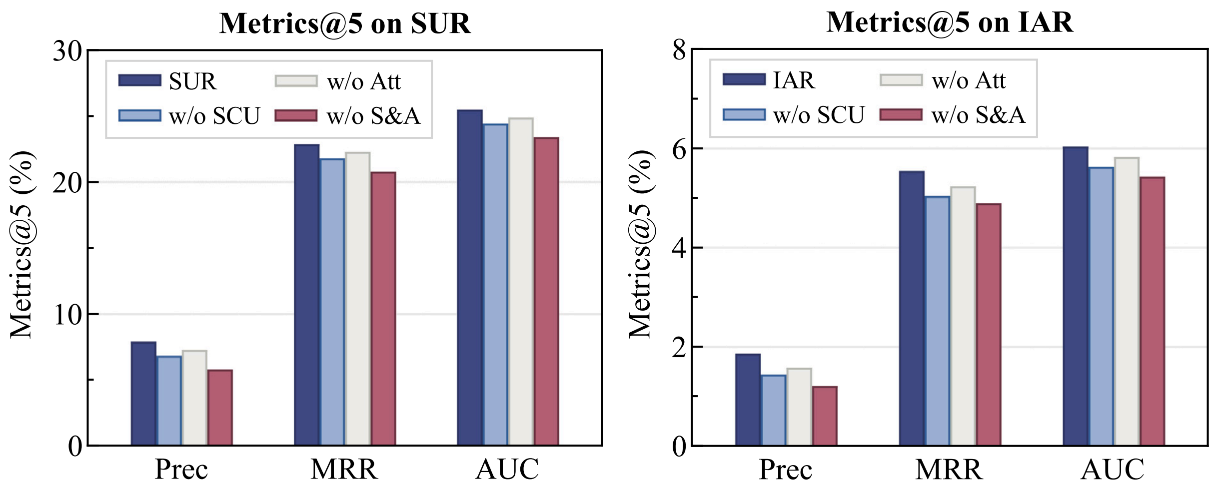


Fig. 9. Effect of SCU and graph attention mechanism.

Table 4
Effect of knowledge transferring.

	Prec@K (%)			MRR@K (%)			AUC@K (%)		
	5	10	20	5	10	20	5	10	20
w/o SKG	8.27	5.42	3.21	25.12	26.86	27.58	27.67	39.20	50.49
w/o CKG	8.48	5.44	3.26	25.18	26.76	27.42	28.11	39.68	50.32
w/o S&C	8.16	5.40	3.17	24.28	26.12	26.88	26.86	38.73	50.47
K-BID	**9.03**	**5.71**	**3.29**	**26.70**	**28.33**	**28.94**	**29.81**	**41.54**	**52.23**
%Improve	6.49	4.96	0.92	6.04	5.47	4.93	6.05	4.69	3.45

The retrieval sets of K-BID and three variants are consistent for fairness.

6.5.2. Ablation study of knowledge retrieval

In the matching phase, the knowledge retrieval models mainly comprise the SCU module for knowledge embedding and the graph attention mechanism for graph-signal propagation. Therefore, we disable one or both components to construct the variants of SUR and IAR. Fig. 9 exhibits the ablation results, where ‘w/o SCU’ means the removal of SCU, ‘w/o Att’ denotes the removal of the graph attention mechanism, and ‘w/o S&A’ removes both components. The following observations can be made:

- Both SCU and graph attention mechanisms are important for the knowledge retrieval models. Both components are complementary to each other, and the ‘w/o S&A’ variants consistently have the worst performance.
- The ‘w/o SCU’ variants underperform the ‘w/o Att’ variants. The large-scale knowledge of user profiles and item attributes may account for this phenomenon. The absence of the knowledge embedding process results in ill-informed nodes and additional noise in graph representation learning.

6.5.3. Ablation study of temporal refinement

In the ranking phase, the knowledge transferring operation empowers the temporal prediction model to be initialized with pretrained embeddings. How different transferred knowledge contributes to the temporal refinement process is an intriguing question. Table 4 records the ablation results with three variants of: (1) ‘w/o SKG’ excluding $\mathbf{U}^{\text{social}}$ learned from SUR; (2) ‘w/o CKG’ excluding $\{\mathbf{U}^{\text{pref}}, \mathbf{V}^{\text{pref}}\}$ learned from IAR; (3) ‘w/o S&C’ excluding all pretrained embeddings. The ablation results demonstrate the necessity of knowledge transferring and accord with the observation in the overall comparison, *i.e.*, both the social and preference factors are complementary and take a considerable effect on information diffusion.

Moreover, to examine the effectiveness of the short-term preference representation and self-attention mechanism, we conduct the ablation experiments as shown in Fig. 10. In comparison with ‘w/o Pref’, the ‘w/o Self’ variant has a minor impact on performance, which suggests a weak capacity to perceive the relevance between features $\mathbf{u}_i^{\text{inv}}, \mathbf{v}_j^{\text{share}}, \mathbf{u}_k^{\text{vot}}$ and $\mathbf{z}_k$. For clarity, we visualize the distribution of voter embeddings in $\mathbb{R}^2$ through Gaussian Kernel Density Estimation (KDE) (Botev, Grotowski, & Kroese, 2010). According to the analysis of previous work (Lin, Tian, Hou, & Zhao, 2022), a more uniform distribution can better preserve the feature diversity and model generalizability. Although the uniform distribution of K-BID is more notable than that of ‘w/o Self’, the pursuit of uniformity may weaken the model to learn close social links among users (or close user–item preference relevance). Contrastive learning may address with this issue, and we leave the trade-off between the social and preference factors as future research endeavour.

6.5.4. Hyper-parameter sensitivity

In the matching phase, the number of graph-signal propagation layers N_l is a pivotal hyper-parameter that affects final vector representations of users and items. We respectively perform sensitivity experiments on SUR and IAR with N_l varying in $\{1, 2, 3, 4\}$. As shown in Fig. 11, the experimental results of SUR illustrate that only one GNN layer is insufficient to aggregate neighbour information, and stacking three layers is the most appropriate choice considering model performance and convergence speed. In contrast, IAR does not benefit from more GNN layers, and one layer is sufficient to capture user-item interactions and achieve fast convergence.

In the ranking phase, because we crop top-K results to measure the metrics, larger retrieval sets appending more similar users and item audience are redundant. A more influential hyper-parameter is the length of voting history $V_{u_k}^{<t_k}$. We search the range of $|V_{u_k}^{<t_k}|$ in $\{2, 5, 10, 15, 20\}$, and corresponding results are shown in Fig. 12. The model reaches the best performance when $|V_{u_k}^{<t_k}| = 5$, and the performance gradually decreases when GRU models longer voting history.

6.5.5. Case study

To demonstrate how temporal changes influence ranking predictions, we conducted a case study by selecting an inviter with median interaction frequency and tracking ten associated users over ten weeks. Using the attention mechanism, we compute the relevance scores between user-item and user-user pairs. The higher the relevance score, the more likely the current user is to be a voter. As visualized in Fig. 13, the intensity of the colour indicates the magnitude of the predicted score, with blue indicating lower scores and red indicating higher scores. The figure shows that voters show different preferences in different periods. $Voter_{13930}$ showed a strong and consistent interest in the items shared by the inviters, indicating a high level of engagement, while

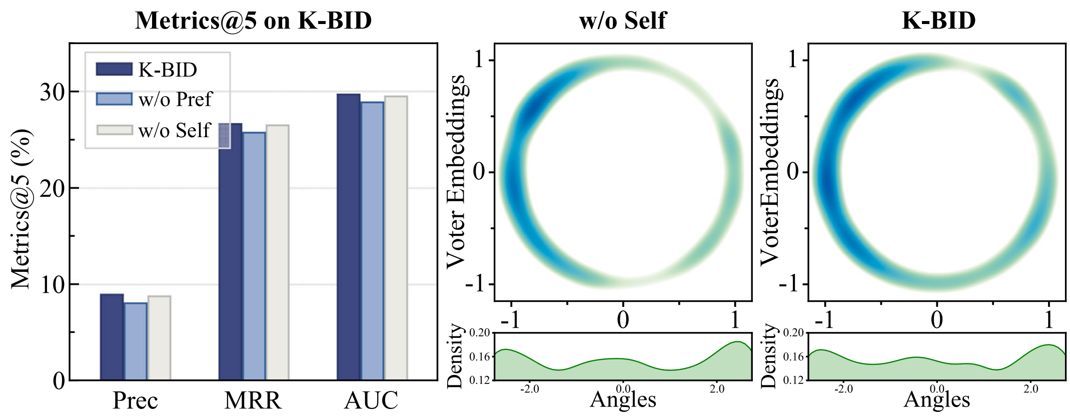


Fig. 10. Effect of GRU and self-attention mechanism in the ranking phase. In the visualization of voter embeddings, darker areas denote a higher density of vectors, and the angular density also reflects whether vectors are concentrated in certain areas.

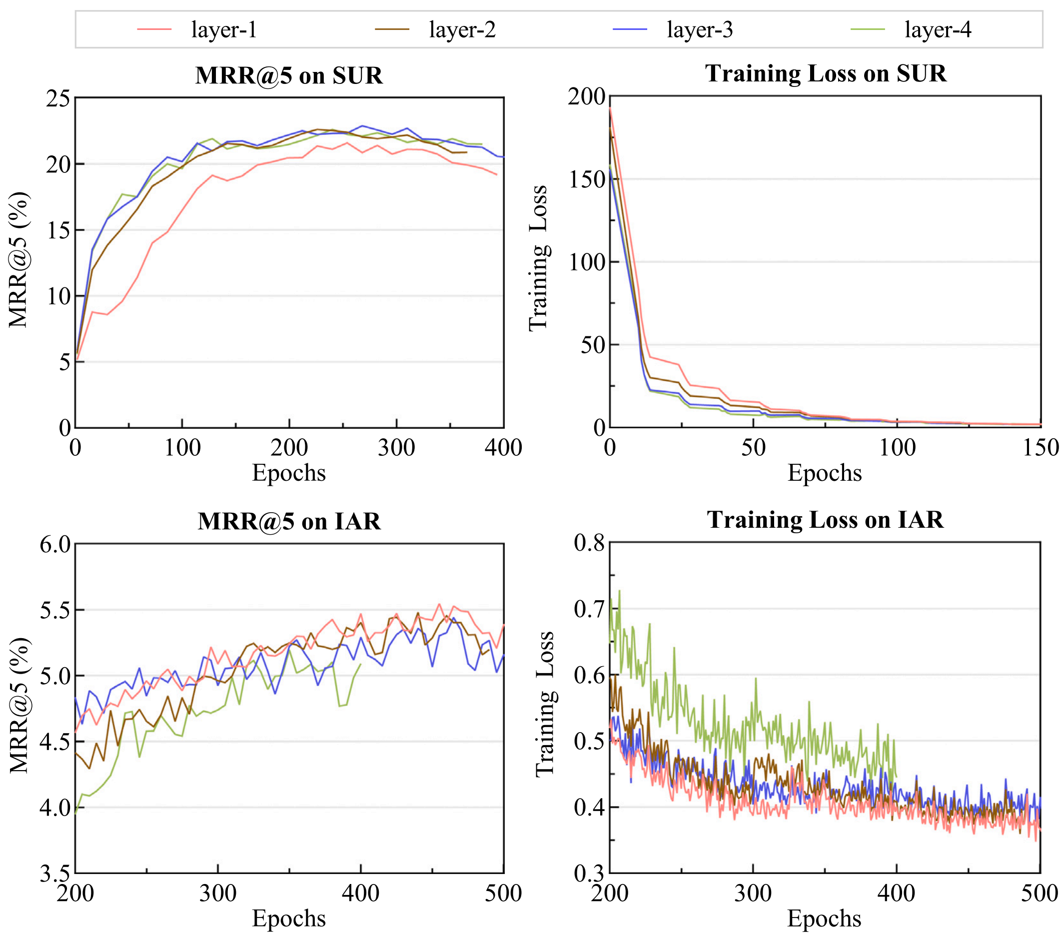


Fig. 11. Effect of the GNN layer number in the matching phase.

$voter_{31957}$ showing the opposite trend. In addition, the behaviour of $voter_{26999}$ is more variable, indicating that their interest in shared items is influenced by specific factors (e.g., social relationships or short-term preferences). This verifies the rationality and necessity of discussing the impact of time changes on ranking prediction at a fine-grained level.

7. Conclusion

In this paper, we have proposed the novel knowledge-driven broad information diffusion model (K-BID) to predict voter behaviour in social networks. Our model leverages semantic cross units and graph attention networks to capture user interests and social relationships effectively, thereby addressing the challenges of graph sparsity and preference ambiguity. Empirical evaluations on large-scale real-world datasets adequately demonstrate the feasibility and effectiveness of K-BID compared with existing methods. This work contributes to the field by advancing the understanding of broad information diffusion and providing a robust framework for predicting user interactions. Theoretical contributions include the integration of semantic and social graph information, offering new insights into the dynamics of information diffusion. Practically, our model offers actionable strategies for companies to enhance user engagement and optimize information dissemination.

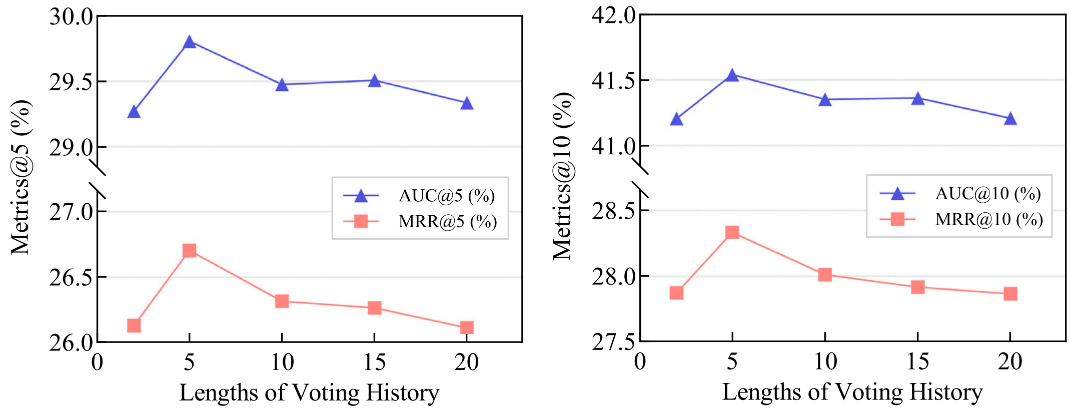


Fig. 12. Effect of the voting history length in the ranking phase.

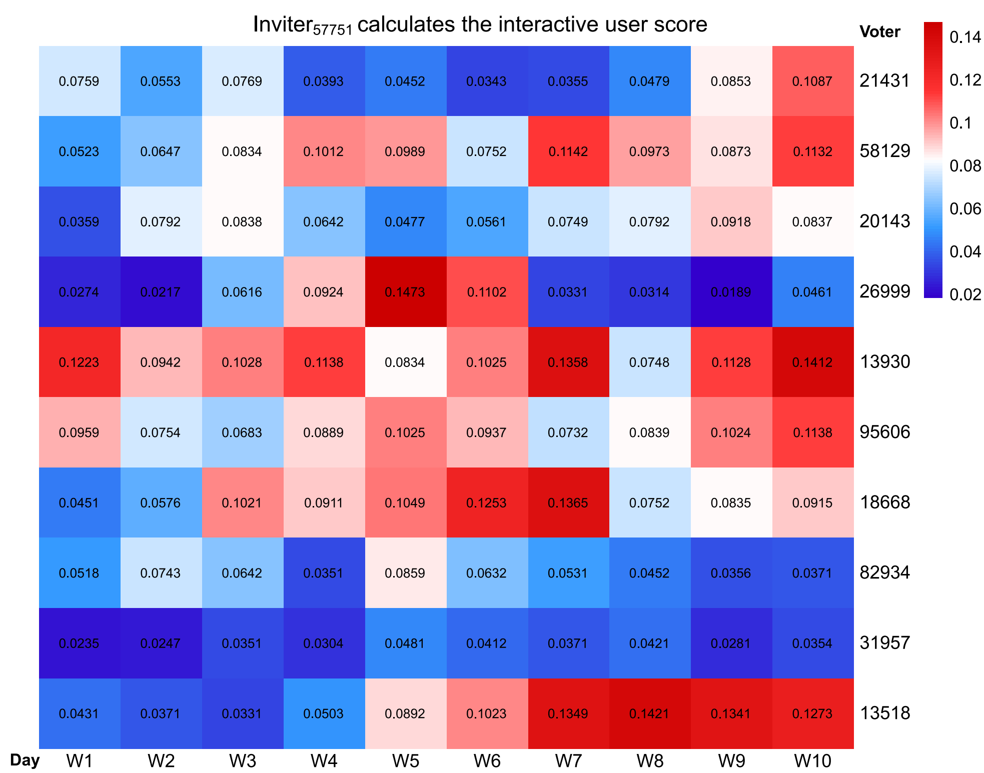


Fig. 13. Interactive user score for $inviter_{57751}$.

Considering the limitations of K-BID, in future research, we plan to further explore explicit modelling of node roles to enhance model performance and interpretability. This may involve incorporating additional data sources, developing more sophisticated role detection algorithms, and incorporating role information into frameworks. We also aim to investigate advanced graph neural network architectures and self-supervised learning techniques to further improve the robustness and generalizability of our framework.

CRediT authorship contribution statement

Xiangjie Kong: Conceptualization, Methodology, Investigation, Funding acquisition, Writing – review & editing. **Can Shu:** Methodology, Data curation, Validation, Writing – original draft. **Lingyun Wang:** Methodology, Data curation, Validation, Writing – review & editing. **Hanlin Zhou:** Conceptualization, Validation, Investigation. **Linan Zhu:** Conceptualization, Investigation, Writing – review & editing, Funding acquisition. **Jianxin Li:** Supervision, Validation, Investigation.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 62476247 and 62072409, in part by the “Pioneer” and “Leading Goose” R&D Program of Zhejiang under Grant 2024C01214, and in part by the Zhejiang Provincial Natural Science Foundation, China under Grant LR21F020003.

Data availability

The source code of K-BID can be found in https://github.com/CanShu6/K-BID.

References

Bordes, A., Usunier, N., Garcia-Duran, A., Weston, J., & Yakhnenko, O. (2013). Translating embeddings for modeling multi-relational data. *Advances in Neural Information Processing Systems*, *26*.

Botev, Z., Grotowski, J., & Kroese, D. (2010). Kernel density estimation via diffusion. *Annals of Statistics*, *38*, 2916–2957. http://dx.doi.org/10.1214/10-aos799.

Bourigault, S., Lamprier, S., & Gallinari, P. (2016). Representation learning for information diffusion through social networks: an embedded cascade model. In *Proceedings of the 9th ACM international conference on web search and data mining* (pp. 573–582). http://dx.doi.org/10.1145/2835776.2835817.

Celma Herrada, Ò. (2010). *Music recommendation and discovery: the long tail, long fail, and long play in the digital music space* (1st ed.). Springer Publishing Company, Incorporated.

Chen, Y., Xie, T., Chen, H., Huang, X., Cui, N., & Li, J. (2024). Kgcf: Social relationship-aware graph collaborative filtering for recommendation. *Information Sciences*, *680*, Article 121102. http://dx.doi.org/10.1016/j.ins.2024.121102.

Chen, J., Xin, X., Liang, X., He, X., & Liu, J. (2023). Gdsrec: Graph-based decentralized collaborative filtering for social recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *35*, 4813–4824. http://dx.doi.org/10.1109/TKDE.2022.3153284.

de Souza, P. R., & Durao, F. A. (2024). Exploiting social capital for improving personalized recommendations in online social networks. *Expert Systems with Applications*, *246*, http://dx.doi.org/10.1016/j.eswa.2023.123098.

Gao, M., Li, J. Y., Chen, C. H., Li, Y., Zhang, J., & Zhan, Z. H. (2023). Enhanced multi-task learning and knowledge graph-based recommender system. *IEEE Transactions on Knowledge and Data Engineering*, http://dx.doi.org/10.1109/TKDE.2023.3251897.

Guo, Q., Zhuang, F., Qin, C., Zhu, H., Xie, X., et al. (2022). A survey on knowledge graph-based recommender systems. *IEEE Transactions on Knowledge and Data Engineering*, *34*, 3549–3568. http://dx.doi.org/10.1109/tkde.2020.3028705.

He, X., & Chua, T. S. (2017). Neural factorization machines for sparse predictive analytics. In *Proceedings of the 40th international ACM SIGIR conference on research and development in information retrieval* (pp. 355–364). http://dx.doi.org/10.1145/3077136.3080777.

Hu, J., Hooi, B., Qian, S., Fang, Q., & Xu, C. (2024). Mgdcf: Distance learning via markov graph diffusion for neural collaborative filtering. *IEEE Transactions on Knowledge and Data Engineering*, *36*, 3281–3296. http://dx.doi.org/10.1109/TKDE.2023.3348537.

Huang, Z., Jin, B., Zhao, H., Liu, Q., Lian, D., et al. (2023). Personal or general? a hybrid strategy with multi-factors for news recommendation. *ACM Transactions on Information Systems*, *41*, http://dx.doi.org/10.1145/3555373.

Huang, C., Yu, F., Wan, Z., Li, F., Ji, H., & Li, Y. (2024). Knowledge graph confidence-aware embedding for recommendation. *Neural Networks*, *180*, Article 106601. http://dx.doi.org/10.1016/j.neunet.2024.106601.

Jiangzhou, D., Songli, W., Jianmei, Y., Lianghao, J., & Yong, W. (2024). Dgrm: Diffusion-gan recommendation model to alleviate the mode collapse problem in sparse environments. *Pattern Recognition*, *155*, Article 110692. http://dx.doi.org/10.1016/j.patcog.2024.110692.

Khaledian, N., Nazari, A., Khamforoosh, K., Abualigah, L., & Javaheri, D. (2023). Trustdl: Use of trust-based dictionary learning to facilitate recommendation in social networks. *Expert Systems with Applications*, *228*, Article 120487. http://dx.doi.org/10.1016/j.eswa.2023.120487.

Kong, X., Chen, Z., Li, J., Bi, J., & Shen, G. (2024a). Kgnext: Knowledge-graph-enhanced transformer for next poi recommendation with uncertain check-ins. *IEEE Transactions on Computational Social Systems*, *11*, 6637–6648. http://dx.doi.org/10.1109/TCSS.2024.3396506.

Kong, X., Mao, M., Wang, W., Liu, J., & Xu, B. (2021). Voprec: Vector representation learning of papers with text information and structural identity for recommendation. *IEEE Transactions on Emerging Topics in Computing*, *9*, 226–237. http://dx.doi.org/10.1109/TETC.2018.2830698.

Kong, X., Shen, Z., Wang, K., Shen, G., & Fu, Y. (2024b). Exploring bus stop mobility pattern: A multi-pattern deep learning prediction framework. *IEEE Transactions on Intelligent Transportation Systems*, *25*, 6604–6616. http://dx.doi.org/10.1109/TITS.2023.3345872.

Kong, X., Wang, K., Hou, M., Xia, F., Karmakar, G., & Li, J. (2022). Exploring human mobility for multi-pattern passenger prediction: A graph learning framework. *IEEE Transactions on Intelligent Transportation Systems*, *23*, 16148–16160. http://dx.doi.org/10.1109/TITS.2022.3148116.

Li, C., Fu, J., Yan, Y., Zhao, Z., & Zeng, Q. (2024). Higher order heterogeneous graph neural network based on node attribute enhancement. *Expert Systems with Applications*, *238*, Article 122404. http://dx.doi.org/10.1016/j.eswa.2023.122404.

Li, N., Guo, B., Liu, Y., & Yu, Z. (2024). Disentangled-feature and composite-prior vae on social recommendation for new users. *Expert Systems with Applications*, *247*, Article 123309. http://dx.doi.org/10.1016/j.eswa.2024.123309.

Li, D., Wang, W., Jin, C., Ma, J., Sun, X., et al. (2019). User recommendation for promoting information diffusion in social networks. *Physica A. Statistical Mechanics and its Applications*, *534*, Article 121536. http://dx.doi.org/10.1016/j.physa.2019.121536.

Liao, L., He, X., Zhang, H., & Chua, T. S. (2018). Attributed social network embedding. *IEEE Transactions on Knowledge and Data Engineering*, *30*, 2257–2270. http://dx.doi.org/10.1109/tkde.2018.2819980.

Lin, Y., Liu, Z., Sun, M., Liu, Y., & Zhu, X. (2015). Learning entity and relation embeddings for knowledge graph completion. In *Proceedings of the AAAI conference on artificial intelligence*. http://dx.doi.org/10.1609/aaai.v29i1.9491.

Lin, Z., Tian, C., Hou, Y., & Zhao, W. X. (2022). Improving graph collaborative filtering with neighborhood-enriched contrastive learning. In *Proceedings of the web conference* (pp. 2320–2329). http://dx.doi.org/10.1145/3485447.3512104.

Liu, Y., Yang, S., Xu, Y., Miao, C., Wu, M., et al. (2023). Contextualized graph attention network for recommendation with item knowledge graph. *IEEE Transactions on Knowledge and Data Engineering*, *35*, 181–195. http://dx.doi.org/10.1109/tkde.2021.3082948.

Lyu, Z., Wu, Y., Lai, J., Yang, M., Li, C., & Zhou, W. (2022). Knowledge enhanced graph neural networks for explainable recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *35*, 4954–4968. http://dx.doi.org/10.1109/tkde.2022.3142260.

Ma, Y., Wang, Z., Tang, M., Qu, W., Deveci, M., & Pamucar, D. (2024). Personalized recommendation system based on meta-learning for human-centric consumer services in gig economy. *IEEE Transactions on Consumer Electronics*, *70*, 1080–1091. http://dx.doi.org/10.1109/TCE.2023.3327408.

Mao, S., & Sejdić, E. (2023). A review of recurrent neural network-based methods in computational physiology. *IEEE Transactions on Neural Networks and Learning Systems*, *34*, 6983–7003. http://dx.doi.org/10.1109/tnnls.2022.3145365.

Opsahl, T., & Panzarasa, P. (2009). Clustering in weighted networks. *Social Networks*, *31*, 155–163. http://dx.doi.org/10.1016/j.socnet.2009.02.002.

Rendle, S., Freudenthaler, C., Gantner, Z., & Schmidt-Thieme, L. (2009). BPR: Bayesian personalized ranking from implicit feedback. In *Proceedings of the 25th conference on uncertainty in artificial intelligence* (pp. 452–461). http://dx.doi.org/10.48550/arxiv.1205.2618.

Shani, G., & Gunawardana, A. (2011). *Evaluating recommendation systems* (pp. 257–297). Boston, MA: Springer US, http://dx.doi.org/10.1007/978-0-387-85820-3_8.

Tong, T., Yuan, W., Jalili, M., Dong, Q., & Sun, J. (2024). A novel ranking approach for identifying crucial spreaders in complex networks based on tanimoto correlation. *Expert Systems with Applications*, *255*, Article 124513. http://dx.doi.org/10.1016/j.eswa.2024.124513.

Wang, X., He, X., Cao, Y., Liu, M., & Chua, T. S. (2019). KGAT: Knowledge graph attention network for recommendation. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 950–958). http://dx.doi.org/10.1145/3292500.3330989.

Wang, X., Huang, T., Wang, D., Yuan, Y., Liu, Z., et al. (2021). Learning intents behind interactions with knowledge graph for recommendation. In *Proceedings of the web conference* (pp. 878–887). http://dx.doi.org/10.1145/3442381.3450133.

Wang, H., Xu, Y., Yang, C., Shi, C., Li, X., et al. (2023). Knowledge-adaptive contrastive learning for recommendation. In *Proceedings of the 16th ACM international conference on web search and data mining* (pp. 535–543). http://dx.doi.org/10.1145/3539597.3570483.

Wang, Z., Zhang, J., Feng, J., & Chen, Z. (2014). Knowledge graph embedding by translating on hyperplanes. In *Proceedings of the AAAI conference on artificial intelligence*. http://dx.doi.org/10.1609/aaai.v28i1.8870.

Wang, H., Zhao, M., Xie, X., Li, W., & Guo, M. (2019). Knowledge graph convolutional networks for recommender systems. In *Proceedings of the web conference* (pp. 3307–3313). http://dx.doi.org/10.1145/3308558.3313417.

Wu, L., Li, J., Sun, P., & Wang, M. (2020). DiffNet++: A neural influence and interest diffusion network for social recommendation. *IEEE Transactions on Knowledge and Data Engineering*, *34*, http://dx.doi.org/10.1109/tkde.2020.3048414.

Wu, L., Sun, P., Fu, Y., Hong, R., Wang, X., et al. (2019). A neural influence diffusion model for social recommendation. In *Proceedings of the 42nd international ACM SIGIR conference on research and development in information retrieval* (pp. 235–244). http://dx.doi.org/10.1145/3331184.3331214.

Wu, S., Sun, F., & Zhang, W. (2023b). Graph neural networks in recommender systems: A survey. *ACM Computing Surveys*, *55*, http://dx.doi.org/10.1145/3535101.

Wu, B., Zhong, L., & Ye, Y. (2023a). Graph-augmented social translation model for next-item recommendation. *IEEE Transactions on Industrial Informatics*, *19*, 10913–10922. http://dx.doi.org/10.1109/tii.2023.3242809.

Xu, G., & Dong, C. (2024). Cagm: A communicability-based adaptive gravity model for influential nodes identification in complex networks. *Expert Systems with Applications*, *235*, Article 121154. http://dx.doi.org/10.1016/j.eswa.2023.121154.

Yang, Y., Huang, C., Xia, L., & Li, C. (2022). Knowledge graph contrastive learning for recommendation. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval* (pp. 1434–1443). http://dx.doi.org/10.1145/3477495.3532009.

Yang, C., Sun, M., Liu, H., Han, S., Liu, Z., & Luan, H. (2021). Neural diffusion model for microscopic cascade study. *IEEE Transactions on Knowledge and Data Engineering*, *33*, 1128–1139. http://dx.doi.org/10.1109/tkde.2019.2939796.

Zhai, S., Liu, B., Yang, D., & Xiao, Y. (2023). Group buying recommendation model based on multi-task learning. In *2023 IEEE 39th international conference on data engineering* (pp. 978–991). http://dx.doi.org/10.1109/ICDE55515.2023.00080.

Zhang, W., Kong, X., Shen, G., Wu, Z., & Qu, Y. (2024). Federated adaptive pseudo-labeling selection for semi-supervised air writing recognition systems. *IEEE Transactions on Consumer Electronics.*, http://dx.doi.org/10.1109/TCE.2024.3467125, 1–1.

Zhang, X., Wang, Z., Liu, G., & Wang, Y. (2024). Key node identification in social networks based on topological potential model. *Computer Communications*, *213*, 158–168. http://dx.doi.org/10.1016/j.comcom.2023.11.003.

Zhang, F., Yuan, N. J., Lian, D., Xie, X., & Ma, W. Y. (2016). Collaborative knowledge base embedding for recommender systems. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery & data mining* (pp. 353–362). http://dx.doi.org/10.1145/2939672.2939673.

Zhao, Z., Hu, A., Zhang, N., Xie, J., Du, Z., Wan, L., et al. (2024). Mining node attributes for link prediction with a non-negative matrix factorization-based approach. *Knowledge-Based Systems*, *299*, Article 112045. http://dx.doi.org/10.1016/j.knosys.2024.112045.

Zhao, Y., Wang, X., Chen, J., Wang, Y., Tang, W., He, X., et al. (2022). Time-aware path reasoning on knowledge graph for recommendation. *ACM Transactions on Information Systems*, *41*, http://dx.doi.org/10.1145/3531267.

Zhou, W., Haq, A. U., Qiu, L., & Akbar, J. (2024). Multi-view social recommendation via matrix factorization with sub-linear convergence rate. *Expert Systems with Applications*, *237*, Article 121687. http://dx.doi.org/10.1016/j.eswa.2023.121687.

Zhou, G., Mou, N., Fan, Y., Pi, Q., Bian, W., Zhou, C., et al. (2019). Deep interest evolution network for click-through rate prediction. In *Proceedings of the AAAI conference on artificial intelligence* (pp. 5941–5948). http://dx.doi.org/10.1609/aaai.v33i01.33015941.

Zhu, J., Dai, Q., Su, L., Ma, R., Liu, J., et al. (2022). BARS: Towards open benchmarking for recommender systems. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval* (pp. 2912–2923). http://dx.doi.org/10.48550/arxiv.2205.09626.

Zou, D., Wei, W., Wang, Z., Mao, X., Zhu, F., et al. (2022). Improving knowledge-aware recommendation with multi-level interactive contrastive learning. In *Proceedings of the 31st ACM international conference on information and knowledge management* (pp. 2817–2826). http://dx.doi.org/10.1145/3511808.3557358.

Dr. Xiangjie Kong is currently a Professor in the College of Computer Science & Technology, Zhejiang University of Technology (ZJUT), China. Previously, he is an Associate Professor in School of Software, Dalian University of Technology (DUT), China. He is Co-Director of The Alpha Lab (http://thealphalab.org/), and Founding Director of City Science of Social Computing Lab (http://cssclab.cn/). He has served as Academic Editor of PeeJ Computer Science (2018-), Academic Editor of Wireless Communications and Mobile Computing (2021-), Associate Editor of IEEE Access (2017-), Editor of KSII Transactions on Internet and Information Systems (2016-), Editor of SpringerPlus (2015–2016), (Leading) Guest Editor of several international journals including Pervasive and Mobile Computing (2018), International Journal of Distributed Sernsor Networks (2015), Future Internet (2019) etc., Workshop Chair or PC Member of a number of conferences including ICDM, ICC, IEEE BigData, UIC, COMPLEX NETWORKS, BigScholar, IOP, MobiSPC, etc.. Dr. Kong has authored/co-authored over 130 scientific papers in international journals and conferences including IEEE TKDE, ACM TKDD, IEEE TII, IEEE COMMUN MAG, IEEE TVT, IEEE IOJ, IEEE TSMC, IEEE TETC, IEEE TASE, IEEE TCSS, WWWJ, FGCS, JNCA, AD HOC NETW, IJRNC SCIM, WWW, etc.. 3 of his papers is selected as ESI- Hot Paper (Top 1%), and 13 papers are ESI-Highly Cited Papers (Top 1%). His research has been reported by Nature Index and other medias. He has been invited as Reviewers for numerous prestigious journals including IEEE TMC, IEEE TITS, IEEE TII, IEEE IOTJ, IEEE COMMUN MAG, IEEE WIREl COMMUN, JOI, JASIST, SCIM, TCJ, etc.. Dr. Kong has authored/co-authored three books (in Chinese). He has contributed to the development of 12 copyrighted software systems and 15 filed patents. He has an h-index of 32 an i10-index of 79, and a total of more than 3400 citations to his work according to Google Scholar. His research interests include big data, network science, and computational social science. He is a Senior Member of IEEE and CCF, a Full Member of Sigma Xi, and a Member of ACM.